Fermi National Accelerator Laboratory

Higher-Order Methods for Quantum Simulations

A.T. Sornborger and E.D. Stewart

Fermi National Accelerator Laboratory
P.O. Box 500, Batavia, Illinois 60510

Disclaimer

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, expressed or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

Distribution

Approved for public release; further dissemination unlimited.

Copyright Notification

This manuscript has been authored by Universities Research Association, Inc. under contract No. DE-AC02-76CHO3000 with the U.S. Department of Energy. The United States Government and the publisher, by accepting the article for publication, acknowledges that the United States Government retains a nonexclusive, paid-up, irrevocable, worldwide license to publish or reproduce the published form of this manuscript, or allow others to do so, for United States Government Purposes.

Higher-Order Methods for Quantum Simulations

A. T. Sornborger and E. D. Stewart

NASA/Fermilab Astrophysics Group, Fermi National Accelerator Laboratory, Box 500, Batavia, IL 60510-0500, USA

(September 3, 1998)

To efficiently implement many-particle quantum simulations on quantum computers we develop and present methods for inverting the Campbell-Baker-Hausdorff lemma to 3rd and 4th order in the commutator. That is, we reexpress $\exp\{-i(H_1 + H_2 + \ldots)\Delta t\}$ as a product of factors $\exp(-iH_1\Delta t)$, $\exp(-iH_2\Delta t)$, ... which is accurate to 3rd or 4th order in Δt.

Quantum computers have generated much interest recently, largely due to the result by Shor [1] that they can factor integers in an amount of time that grows polynomially with the size of the integer. This can be compared to factorization on a classical computer, where the time it takes to factor a number grows exponentially with the input size. In addition to Shor's factorization algorithm, simulations of quantum systems have also been shown to be possible in polynomial time [2]. Indeed, this was the first area for which it was proposed that quantum computers could fundamentally be more powerful (i.e. much faster) than classical computers [3].

From a theoretical standpoint, a quantum computer is a quantum system with a 2^n-dimensional Hilbert space. Pairs of states in the system are defined to be 'qubits'. The canonical example of such a system is a set of n spins. Each spin consists of two states, so each spin can represent a qubit and the Hilbert space of the system is 2^n-dimensional. The equivalent of a logical gate on a classical computer is an operator acting on a set of qubits on a quantum computer.

This letter focuses on a problem which concerns simulational issues in quantum computation. A simulation of a quantum mechanical system on a quantum computer consists of applying an operator $\exp(-iHt)$ on a set of qubits, where H, the Hamiltonian of the system of interest, is suitably encoded (and discretized) to act on the set of qubits. For many-particle systems H is a sum of terms. For instance, the Hubbard model Hamiltonian, used in the study of high-T_c superconductivity, can be written [4] as the sum

$$H = \sum_{i=1}^{m} V_0 n_{i\uparrow} n_{i\downarrow} + \sum_{\langle i,j \rangle \sigma} t_0 c_{i\sigma}^* c_{j\sigma} \tag{1}$$

where V_0 is the strength of the potential, and $n_{i\sigma}$ is the operator for the number of fermions of spin σ at site i. In the second (kinetic energy) term, the sum $\langle i,j \rangle$ indicates all neighboring pairs of sites, t_0 is the strength of the "hopping", and $c_{i\sigma}$, $c_{i\sigma}^*$ are annihilation and creation operators, respectively, of a fermion at site i and spin σ. This model gives an example in which a full simulation on a classical computer is impossible due to the exponential increase in the size of the Hilbert space of the quantum system with the number of lattice sites.

The canonical quantum computer cannot act on all spins at once [5]. Therefore, it becomes necessary to find ways of approximating the evolution operator, which is the exponential of a sum of operators (with a Hamiltonian such as that in Eq. (1)) as a product of operators each acting on a subspace of the Hilbert space. To second order, for instance, we could use the approximation

$$
\begin{aligned}
&\left(e^{-iH_1\Delta t} e^{-iH_2\Delta t} \ldots e^{-iH_N\Delta t}\right) \\
&\left(e^{-iH_N\Delta t} \ldots e^{-iH_2\Delta t} e^{-iH_1\Delta t}\right) \\
&= e^{-i2(H_1 + H_2 + \ldots + H_N)\Delta t + \mathcal{O}\left[(\Delta t)^3\right]}
\end{aligned}
\tag{2}
$$

where the $e^{-iH_n\Delta t}$ act on a subspace of the Hilbert space.

To find higher order approximation methods, we want to reexpress $\exp\left(\sum_{n=1}^{N} A_n\right)$ as a product of individual $\exp(A_n)$'s. In order to do this, we must invert the Campbell-Baker-Hausdorff formula. To 5th order, the Campbell-Baker-Hausdorff formula is

$$
\begin{aligned}
&\exp(aA_1)\exp(aA_2) = \\
&\exp\Big[a(A_1 + A_2) + \frac{1}{2}a^2 A_{12} + \frac{1}{12}a^3(A_{112} + A_{221}) \\
&+ \frac{1}{24}a^4 A_{1221} - \frac{1}{720}a^5(A_{11112} - 2A_{21112} - 6A_{11221} \\
&- 6A_{22112} - 2A_{12221} + A_{22221}) + \mathcal{O}(a^6)\Big]
\end{aligned}
\tag{3}
$$

where

$$A_{kl\ldots mn} \equiv [A_k, [A_l, \ldots [A_m, A_n]\ldots]] \tag{4}$$

As a strategy for finding approximation methods, we pick a fundamental ordering of the product of exponentials with parameters allowing for transposes of the entire product as well as raising all the exponentials in the fundamental unit to the same power. By iterating the Campbell-Baker-Hausdorff formula, we can get an expression for this fundamental unit in terms of a single exponential

$$\left(e^{aA_1} e^{aA_2} \ldots e^{aA_N}\right)^{\alpha} = \exp \sum_{p=1}^{\infty} \alpha a^p B_N^p \tag{5}$$

which defines the B_N^p in terms of the A_n. Here, p is an exponent on a, and a label on the matrices B_N^p.

Now combine a succession $i = 1, \ldots, I$ of fundamental units with parameters a_i and α_i. Again iterating Campbell-Baker-Hausdorff gives

$$\exp \left(\sum_{p=1}^{\infty} \alpha_1 a_1^p B_N^p \right) \ldots \exp \left(\sum_{p=1}^{\infty} \alpha_I a_I^p B_N^p \right)$$
$$= \exp \left(\sum_X \sigma_I^X B_N^X \right) \qquad (6)$$

The B_N^X are generated from the B_N^p by commutation. X represents a label $pq \ldots rs$ where

$$B_N^{pq \ldots rs} \equiv [B_N^p, [B_N^q, \ldots [B_N^r, B_N^s] \ldots]] \qquad (7)$$

$B_N^{pq \ldots rs}$ is of order $p + q + \ldots + r + s$. Up to 5th order we can take

$$X \in \{1; 2; 3, 12; 4, 13, 112; 5, 14, 23, 113, 221, 1112\} \quad (8)$$

These B_N^X span the space of the B_N^p's and their commutators to 5th order and for $N \geq 2$ they are independent. The σ_I^X are defined in terms of α_i and a_i by Eq. (6). Here again, the X's are labels. After some calculation, the Campbell-Baker-Hausdorff formula, Eq. (3), gives the equations

$$\sigma_I^p = \sum_{i=1}^{I} \alpha_i a_i{}^p \qquad (9)$$

for $p = 1, \ldots, 5$,

$$\sigma_I^{pq} = -\frac{1}{2} \sigma_I^p \sigma_I^q + \frac{1}{2} \sum_{i=1}^{I} a_i{}^{q-p} \left[\left(\sigma_i^p \right)^2 - \left(\sigma_{i-1}^p \right)^2 \right] \quad (10)$$

for $pq = 12, 13, 14, 23$,

$$\sigma_I^{ppq} = -\frac{1}{2} \sigma_I^p \sigma_I^{pq} - \frac{1}{6} \left(\sigma_I^p \right)^2 \sigma_I^q$$
$$+ \frac{1}{6} \sum_{i=1}^{I} a_i{}^{q-p} \left[\left(\sigma_i^p \right)^3 - \left(\sigma_{i-1}^p \right)^3 \right] \qquad (11)$$

for $ppq = 112, 113, 221$, where $\sigma_I^{21} \equiv -\sigma_I^{12}$

$$\sigma_I^{1112} = -\frac{1}{2} \sigma_I^1 \sigma_I^{112} - \frac{1}{3} \left(\sigma_I^1 \right)^2 \sigma_I^{12} - \frac{1}{24} \left(\sigma_I^1 \right)^3 \sigma_I^2$$
$$+ \frac{1}{24} \sum_{i=1}^{I} a_i \left[\left(\sigma_i^1 \right)^4 - \left(\sigma_{i-1}^1 \right)^4 \right] \qquad (12)$$

For approximations to $\exp \left(\sum_{n=1}^{N} A_n \right)$, we require all $\sigma_I^X = 0$ except for σ_I^1 which is the coefficient of $B_N^1 = \sum_{n=1}^{N} A_n$, and which should be greater than zero.

An interesting feature of 3rd order methods is that they require at least one inverse, i.e. they require backward time evolution during part of the method.[*] This follows immediately from Eq. (9) with $p = 3$. It can also be proved using Eq. (9) with $p = 3$ and $p = 4$ that 4th order methods must have at least two inverses.

Our basic method to solve Eqs. (9-11) is to pick values of α_i and a_i and see if they satisfy the equations. To do this we must restrict the number of fundamental units by fixing I. We also take the α_i's to be ± 1 and restrict the range of the a_i's.

We start with Eq. (9), since, in this equation, order with respect to i does not matter. So, for a given set of values, we need to consider only one permutation, not all permutations of the values. This greatly reduces the size of the search.

Furthermore, we start by considering $p = 1$ and 3, since it is only the sign of $\alpha_i a_i$ that matters in these equations. This means we can consider only the sign of the combination $\alpha_i a_i$, and not the signs of α_i and a_i individually. This reduces the search further. These equations are particularly restrictive for the case of few inverses.

After solving the $p = 1$ and 3 equations, we introduce separate signs for the α_i's and a_i's and solve the equation with $p = 2$, and $p = 4$ for the 4th order case.

Finally, into the restricted set of solutions to Eq. (9) we introduce permutations of the α_i's and a_i's with respect to the index i and solve Eqs. (9-11).

We find a larger number of solutions than we can easily present. We want to present solutions which are in some sense optimal. To do this, we consider the form of the operator resulting from a given method

$$\prod_{j=1}^{I} \left(e^{-i a_j A_1 \Delta t} e^{-i a_j A_2 \Delta t} \ldots e^{-i a_j A_N \Delta t} \right)^{\alpha_j}$$
$$= \exp \left[-i \sigma_I^1 \sum_{n=1}^{N} A_n \Delta t + r(-i \Delta t)^{o+1} \right] \qquad (13)$$

where $\Delta t \ll 1$ is a time step, o is the order of the method, and

$$r = \sum_X \sigma_I^X B_N^X \qquad (14)$$

where $X \in \{4, 13, 112\}$ for a 3rd order method and $X \in \{5, 14, 23, 113, 221, 1112\}$ for a 4th order method.

r is an error which takes values in the vector space of the commutators for which we do not have a metric. Therefore, we make an *ad hoc* choice of basis to be discussed elsewhere. This allows us to replace r by a single real scalar R. The error from the method can then be taken to be

---

[*]After this work was completed, we became aware that this point had also been noted in [6].

$$E = nR\,\Delta t^{o+1} \tag{15}$$

where n is the number of times we apply the approximate method.

If the physical time we want to simulate is T_p, then

$$T_p = nD\Delta t \tag{16}$$

where $D \equiv \sigma_I^1$ is given by the method.

The computer time it takes for a given simulation can be written

$$T_c = nINt_g + nLNt_s \tag{17}$$

where I is the number of fundamental units in the method and N is the number of terms in a unit, t_g is the time it takes to make a gate change,

$$L \equiv \sum_{i=1}^{I} |a_i| \tag{18}$$

so that LN is the total time the gates are applied for in the method. The time an individual gate is applied for will be $t_s = b\,\Delta t$, where b is a proportionality constant dictated by the actual couplings in the quantum computer hardware.

Using Eqs. (15) and (16), the computer time can be rewritten

$$T_c = \left(\frac{T_p^{o+1}}{E}\right)^{\frac{1}{o}} \left(\frac{G}{D}\right) \left(\frac{R}{D}\right)^{\frac{1}{o}} t_g + \frac{LbT_p}{D} \tag{19}$$

There are two possible limits to this equation. One is that the computer time is dominated by gate switching. In this case, we want the factor

$$Z = \left(\frac{G}{D}\right) \left(\frac{R}{D}\right)^{\frac{1}{o}} \tag{20}$$

to be small. The second limit is when the computer time is dominated by the time during which the gates are applied. Here, the ratio L/D should be small, and to make the error small, we want $(R/D)(\Delta t)^o$ small. However, if Δt can be made very small (from the hardware point of view), then making the error small forces the computer time to be dominated by gate switching. If there is a limit to Δt, and it is reached before the computer time is gate switching dominated, then the computer time may still be dominated by gate application and we want L/D and R/D small. We also prefer to have concise methods.

The 3rd order method that we have selected given the above criteria is

$$(1)^T(1)(1)(1)(1)^T(-2)^T(1)(1)(1) \tag{21}$$

and the 4th order method is

$$(1)^T(1)(1)^T(-2)(1)^T(1)^T(1)^T(1)^T(1)$$
$$(1)^T(1)(1)(1)(1)(-2)^T(1)(1)^T(1) \tag{22}$$

where (x) denotes

$$(e^{xA_1}e^{xA_2}\dots e^{xA_N}) \tag{23}$$

and $(x)^T$ denotes

$$(e^{xA_N}\dots e^{xA_2}e^{xA_1}) \tag{24}$$

To illustrate our methods, we have applied first, second, third and fourth order methods to the exactly soluble operator

$$e^{-i\,\Delta t(\sigma_x + \sigma_y + \sigma_z)} = \begin{pmatrix} C - \frac{i}{\sqrt{3}}S & -\frac{1+i}{\sqrt{3}}S \\ \frac{1-i}{\sqrt{3}}S & C + \frac{i}{\sqrt{3}}S \end{pmatrix} \tag{25}$$

where $C \equiv \cos\left(\sqrt{3}\,\Delta t\right)$ and $S \equiv \sin\left(\sqrt{3}\,\Delta t\right)$.

As a measure of the error, we took the differences $\Delta\sigma_x$, $\Delta\sigma_y$ and $\Delta\sigma_z$ between the σ_x, σ_y and σ_z components of the exact solution and those of the results of our methods. We then calculated the error

$$E = \sqrt{(\Delta\sigma_x)^2 + (\Delta\sigma_y)^2 + (\Delta\sigma_z)^2} \tag{26}$$

In Fig. (1), we plot the logarithm of the error as a function of the logarithm of the time that the system was evolved for. The first order method results are uppermost and higher order results lie underneath each other with fourth order results being the lowermost plotted. $\Delta t = 0.01$ for all methods.
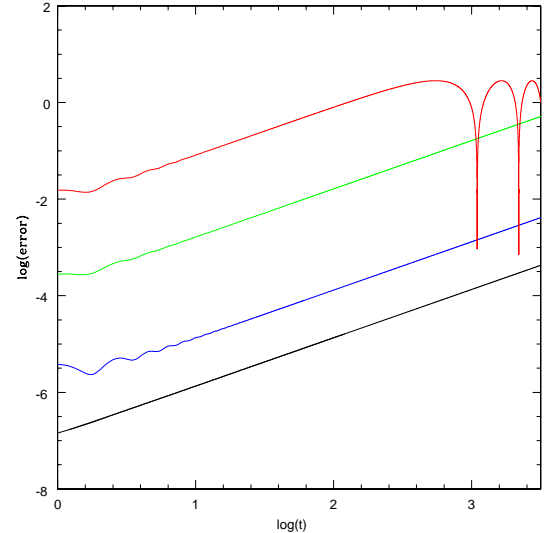

FIG. 1. A measurement of the accuracy of our results. Plotted is the $\log(error)$ of (from top to bottom) first, second, third and fourth order approximation methods as a function of $\log(time)$. Note that for the fourth order method, the error never grows larger than 10^{-3}.

3

Notice that the first order error oscillates once it reaches order 1. The rest of the errors remain small throughout the simulation, with the fourth order error remaining below 10^{-3} for the entire evolution.

The error for all methods goes as $nR\left(\Delta t\right)^{o+1}$, where n is the number of times the method has been applied. Therefore, $\log E = \log n + \log\left[R\left(\Delta t\right)^{o+1}\right]$. For $\Delta t = 0.01$, this makes the y-intercept decrease roughly by order -2 as the order of the method increases. Since the time evolved is proportional to n, the slope of the errors is 1 for all methods.

As an example of how useful our approximations can be, let us consider a case in which we want to apply an approximation method for time $T = 1$ with total error $E = 10^{-4}$. For a first order method, this means that we require about 5000 applications of the method. For second order, we require about 30 applications. For our third order method, we need 2 applications. And for our fourth order method, we need less than 1 application of the method. This results in a reduction of orders of magnitude in the computational cost of a given simulation.

ACKNOWLEDGEMENTS

[1] P. W. Shor, in *Proceedings of the 35th Annual Symposium on Foundations of Computer Science, Santa Fe, NM, 1994*, edited by Shafi Goldwasser (IEEE Computer Society Press, Los Alamitos, CA, 1994), 124-134; SIAM J. Comput. **26**, 1484-1509 (1997).

[2] B. M. Boghosian and W. Taylor IV, quant-ph/9701019; quant-ph/9701016; quant-ph/9604035; D. S. Abrams and S. Lloyd, *Phys. Rev. Lett.* **79**, 2589-2589 (1997).

[3] R. P. Feynman, *Int. Jour. of Theor. Phys.*, Vol. 21, Nos. 6/7, 467-488 (1982).

[4] D. S. Abrams and S. Lloyd, *Phys. Rev. Lett.* **79**, 2586-2589 (1997).

[5] A. Steane, quant-ph/9708022; S. Lloyd, *Phys. Rev. Lett.* **75**, 346-349 (1995); D. Deutsch, A. Barenco and A. Ekert, quant-ph/9505018; A. Barenco, et. al., quant-ph/9503016.

[6] D. Goldman and T. Kaper, *SIAM J. Num. Anal.* **33**, No. 1, 349-367 (1996). M. Suzuki, *Phys. Lett. A* Vol. 146, No. 6, 319-323 (1990); M. Suzuki, *J. Math. Phys.* **32**, No. 2, 400-407 (1991);